UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|---|---|---|---|---|
| 09/915,939 | 07/25/2001 | Amit P. Singh | 1014-170US01 | 8246 |

28863 7590 05/30/2006

SHUMAKER & SIEFFERT, P. A.
8425 SEASONS PARKWAY
SUITE 105
ST. PAUL, MN 55125

| EXAMINER | |
|---|---|
| MEW, KEVIN D | |
| **ART UNIT** | **PAPER NUMBER** |
| 2616 | |

DATE MAILED: 05/30/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

| Office Action Summary | Application No. | Applicant(s) |
|---|---|---|
| | 09/915,939 | SINGH, AMIT P. |
| | Examiner | Art Unit |
| | Kevin Mew | 2616 |

*-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --*

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

1)☒ Responsive to communication(s) filed on *30 November 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

4)☒ Claim(s) *2-15,18-55,58-81 and 83-95* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *2-15,18,19,24-55,58,59,63-81 and 83-95* is/are rejected.

7)☒ Claim(s) *20-23 and 60-62* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

**Application Papers**

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *7/25/2001* is/are: a)☐ accepted or b)☒ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date *20*.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

## *Detailed Action*

### *Response to Amendment*

1. Applicant's Remarks/Arguments filed have been considered. Claims 2-15, 18-55, 58-81, 83-95 are currently pending and claims 1, 16-17, 56-57, 82, 96 have been canceled by applicant.

### *Drawings*

2. The drawings are objected to because the title, application number, inventor name, attorney docket number information shown at the top of each figure should not be included in the drawings. In addition, Fig. 3a of the drawings is objected to because it lacks descriptive labels/legends.

Corrected drawing sheets in compliance with 37 CFR 1.121(d) are required in reply to the Office action to avoid abandonment of the application. Any amended replacement drawing sheet should include all of the figures appearing on the immediate prior version of the sheet, even if only one figure is being amended. The figure or figure number of an amended drawing should not be labeled as "amended." If a drawing figure is to be canceled, the appropriate figure must be removed from the replacement sheet, and where necessary, the remaining figures must be renumbered and appropriate changes made to the brief description of the several views of the drawings for consistency. Additional replacement sheets may be necessary to show the renumbering of the remaining figures. Each drawing sheet submitted after the filing date of an application must be labeled in the top margin as either "Replacement Sheet" or "New Sheet" pursuant to 37 CFR 1.121(d). If the changes are not accepted by the examiner, the applicant will

be notified and informed of any required corrective action in the next Office action. The objection to the drawings will not be held in abeyance.

### *Claim Objections*

3. Claims 26-27, 58, 83 are objected to because of the following informalities:

In claim 26, line 12, remove the repeating phrase "the encoder module."

In claim 27, line 12, remove the repeating phrase "the encoder module."

In claim 58, line 1, the phrase "The system of claim 56" should be amended to "The method of claim 42" because claim 56 has already been canceled by applicant.

In claim 83, line 1, the phrase "The method of claim 82" should be amended to "The method of claim 72" because claim 82 has already been canceled by applicant.

Appropriate correction is required.

### *Claim Rejections - 35 USC § 102*

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 85-87 are rejected under 35 U.S.C. 102(e) as being anticipated by Malcolm et al. (WO 00/07124).

Regarding claim 85, Malcolm discloses a system for decoding one or more repetitive data blocks in data communicated over a network comprising:

transmitting routing information for identifying one or more addresses a decoder module supports over a network to an encoder module **(each web object destined to a leaf cache having a URL address**, page 9, lines 20-26)

an decoder module (**a leaf cache**, element 111, Fig. 1), coupled in the network (**coupled in a communication network**, element 100, Fig. 1), the decoder module intercepting the data (**leaf cache receives object signature from root cache**, page 16, lines 29-30 and Fig. 1), the decoder module receiving data blocks (**object signatures**, page 16, lines 13-14) for different communication sessions (**for communication requests using different protocols such as FTP and HTTP**, page 7, lines 28-32) from a corresponding encoder module (**from the root cache**); and

a memory (**a memory storage**, element 112, Fig. 1), accessible to the decoder module (**accessible to leaf cache**, element 111, Fig. 1), for storing the contents of one or more

data blocks previously transmitted by the encoder module (**for caching for previously transmitted object signatures**, page 17, lines 1-4) wherein the decoder module (**leaf cache**, element 111, Fig. 1) determines whether the contents of each of the received data blocks is in encoded form (**leaf cache determines whether the received data is in object signature form**, page 17, lines 1-4),

wherein responsive to the respective data block being in encoded form (**if web object is in object signature form is received in the leaf cache**, page 16, lines 22-30), the decoder module selects the contents of a matching previously received block as the contents of the respective encoded block and the respective extracted data block (**leaf cache selects the web object of the matching object signature,** page 17, lines 1-4);

wherein responsive to the data block being unencoded, the decoder module stores the contents of the respective received data block as a previously received data block (**when the actual web object is encountered at the leaf cache, the leaf cache stores it to be served to requesting client device**, page17, lines 9-14).

Regarding claims 86, Malcolm discloses the system of claim 85 further comprising receiving an indicator for identifying that the contents of a respective data block have been previously transmitted (**receives a object signature from the root cache for identifying the web object**, page 16, lines 29-30).

Regarding claim 87, Malcolm discloses the method of claim 85 further comprising the identification of the one or more previously received data blocks from a corresponding encoder module (**each of the web objects having a URL as a unique identifier**, page 9, lines 20-26).

### *Claim Rejections - 35 USC § 103*

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 2-4, 6-11, 15, 18, 24-31, 42-43, 45-51, 58, 72-79, 89-91 are rejected under 35 U.S.C. 103(a) as being unpatentable over the admitted prior art, Malcolm et al. (WO 00/07124), in view of Sarkissian et al. (USP 6,771,646).

Regarding claims 2, 72, 76-79, Malcolm discloses a system and method for encoding one or more repetitive data blocks in data communicated over a network comprising:

an encoder module (**a root cache**, element 111, Fig. 1), coupled in the network (**coupled in a communication network**, element 100, Fig. 1), the encoder module intercepting the data (**root cache receives web object from server device**, page 16, lines 10-15 and Fig. 1), the encoder module extracting data blocks (**object signatures**, page 16, lines 13-14) from different communication sessions (**from communication requests using different protocols such as FTP and HTTP**, page 7, lines 28-32) from the intercepted data (**from the web object**), each extracted block (**web object**, page 10-15) having a destination address supported

for decoding by at least one corresponding decoder module (**each web object having a URL address**, page 9, lines 20-26), and the encoder module passing through data not having a supported destination address (**router switch 113 of root cache 111 passes through other types of requests other than the supported web object's URL identifier**, page 7, lines 28-32).

a memory (**a leaf cache**, element 111, Fig. 1), accessible to the encoder module (**accessible to root cache**, element 111, Fig. 1), for storing the contents of one or more data blocks previously transmitted by the encoder module (**a leaf cache for caching for previously transmitted web objects**, page 16, lines 16-20) wherein the encoder module (**root cache**, element 111, Fig. 1) determines whether there is a match between the contents of each of the extracted data blocks and the contents of at least one previously transmitted data block (**root cache determines whether the extracted web object is present in the leaf cache**, page 16, lines 22-27),

wherein responsive to said match (**if web object is present in the leaf cache**, page 16, lines 22-30), the encoder module encodes the respective extracted data block (**the web object is encoded as object signature, page 16**, lines 22-30) and transmits the respective extracted data block in encoded form (**root cache transmits the object signature**) to the at least one corresponding decoder module (**to the leaf cache**, page 16, lines 22-30);

responsive to no match (**if web object is not present at the leaf cache**, page 17, lines 1-5), the encoder module (**root cache**) transmits the respective extracted data block in intercepted form (**root cache transmits the actual web object**, page 17, lines 1-5) to the at least one corresponding decoder module (**to the leaf cache**, page 17, lines 9-11); and

each of the previously transmitted unique data blocks having a unique identifier (**each of the web objects having a URL as a unique identifier**, page 9, lines 20-26).

Malcolm does not explicitly disclose the memory comprises a least recently used data structure for storing one or more previously transmitted unique data blocks, said least recently used data structure having a maximum capacity and a position in an order of most recently used to least recently used of the one or more stored blocks, and

responsive to a match, associates the previously transmitted data block having the matching contents with the position in the least recently used data structure indicating the most recently used previously transmitted data block.

However, Sarkissian discloses a least recently used data structure having a maximum capacity (LRU) (**CAM array/stack**, col. 31, lines 24-50) in a cache system of a packet network in which the addressable memory cells CAMs are ordered according to recentness of use, with the most recently used cache contents pointed to by the top CAM and the least recently used cache contents pointed to by the bottom CAM (col. 31, lines 41-50). Sarkissian further discloses when there is a cache hit/match, the contents of the CAM that produced the hit are put in the top CAM of the stack while the CAM contents of the CAM above the CAM that produced the hit are shifted down to fill the gap (col. 31, lines 32-40).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Sarkissian in employing a least recently used data structure/algorithm such that the leaf cache and root cache disclosed in Malcolm will implement the least recently used data structure of Sarkissian such that the leaf and root caches of Malcolm comprise a least recently used data structure for

storing one or more previously transmitted unique data blocks, said least recently used data structure having a maximum capacity and a position in an order of most recently used to least recently used of the one or more stored blocks, and in response to a match, associates the previously transmitted data block having the matching contents with the position in the least recently used data structure indicating the most recently used previously transmitted data block.

The motivation to do so is to employ the least recently used replacement policy in cache system it replaces the least recently used flow entry when a cache replacement is needed because it is likely that a packet following a recent packet will belong to the same flow and thus the signature of a new packet will likely match a recently used flow record, and it is not highly likely that a packet associated with the least recently used flow-entry will arrive soon.

Regarding claim 3, Malcolm discloses the system of claim 2 wherein the encoded form of the respective extracted data block is transparent to one or more nodes in the network (**a cache may discard a web object that the object signature it receives can no longer be used to determine the web object, meaning the object signature is transparent to that cache**, page 12, lines 23-27).

Regarding claim 4, Malcolm discloses the system of claim 2 wherein the encoder module (**a root cache**, element 111, Fig. 1) is coupled via a switch (**a router-switch**, element 113, Fig. 1) in a physical connection between two nodes of the network (**in a physical connection between a client and a server in a communication network**, elements 120. 130, Fig. 1), responsive to a first configuration of the switch (**responsive to the configuration of processing**

**FTP or HTTP requests**), the encoder module processing data that traverse the physical connection between these two nodes (**the root cache processes FTP or HTTP requests**, page 7, lines 28-32), and responsive to a second configuration of the switch (**responsive to the configuration of processing all other types of requests**), the data bypassing the encoder module (**the cache passes through these data unchanged**, page 7, lines 28-32).

Regarding claims 6, 73, Malcolm discloses the system and method of claims 2, 72 wherein the encoder module, responsive to a match in contents (**root cache, response to the web object being present in the leaf cache**, page 16, lines 22-30), transmits an indicator, identifying that the contents of a respective data block have been previously transmitted (**transmits a object signature identifying the web object**, page 16, lines 22-30).

Regarding claim 7, Malcolm discloses the system of claim 6 wherein the indicator is a special symbol (**object signature is a special symbol, which is a form of dictionary compression**, page 12, lines 29-32).

Regarding claim 8, Malcolm discloses the system of claim 6 wherein the indicator is an extra header (**object signature is a form of dictionary compression of a web object**, page 12, lines 29-32).

Regarding claim 9, Malcolm discloses the system of claim 2 wherein at least one respective data block is a packet payload (**web object includes text or multimedia data**, page 6, lines 4-21).

Regarding claim 10, Malcolm discloses the system of claim 2 wherein at least one respective data block is a portion of a packet payload (**web object is a portion of the packet payload**, page 6, lines 4-21).

Regarding claims 11, 74, Malcolm discloses the system of claims 2, 72 wherein the encoder module encodes at least one extracted data block (**the sender encodes the web object in MD5 signature**) using a synchronization mechanism for verifying the identification of the one or more previously transmitted data blocks with the at least one corresponding decoder module supporting the destination address of the extracted data block (**using this MD5 signature dictionary compression as a synchronization mechanism to verify the identification a previously transmitted web object at the destination**, page 12, lines 6-11, 17-20, 22-27).

Regarding claim 12, Malcolm discloses the system of claim 11 wherein the synchronization mechanism is an explicit synchronization mechanism (**root cache receives an object signature from server device**, page 9, lines 3-8).

Regarding claims 15, 75, Malcolm discloses the system of claims 2, 72 further comprising the encoder module, except fails to disclose responsive to no match between the contents of the respective extracted data block and the contents of at least one of the previously transmitted blocks, determining whether to delete at least one of the previously transmitted data blocks in the memory; and storing the respective extracted data block in intercepted form in the memory.

However, Sarkissian discloses that if there is a cache miss, any new flow record is put in the cache memory element pointed to by the bottom CAM (col. 31, lines 41-42).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Sarkissian in employing a least recently used data structure/algorithm such that the leaf cache and root cache disclosed in Malcolm will implement the least recently used data structure of Sarkissian such that in response to no match for the web object, the LRU data structure of leaf and root caches of Malcolm will determining whether to delete at least one of the previously transmitted data blocks in the memory, and storing the respective extracted data block in intercepted form in the memory. The motivation to do so is to create a new record in memory for this new flow.

Regarding claim 18, Malcolm discloses a system for encoding one or more repetitive data blocks in data communicated over a network comprising:

an encoder module (**a root cache**, element 111, Fig. 1), coupled in the network (**coupled in a communication network**, element 100, Fig. 1), the encoder module intercepting the data (**root cache receives web object from server device**, page 16, lines 10-15 and Fig. 1),

the encoder module extracting data blocks (**object signatures**, page 16, lines 13-14) from different communication sessions (**from communication requests using different protocols such as FTP and HTTP**, page 7, lines 28-32) from the intercepted data (**from the web object**), each extracted block (**web object**, page 10-15) having a destination address supported for decoding by at least one corresponding decoder module (**each web object having a URL address**, page 9, lines 20-26), and the encoder module passing through data not having a supported destination address (**router switch 113 of root cache 111 passes through other types of requests other than the supported web object's URL identifier**, page 7, lines 28-32).

a memory (**a leaf cache**, element 111, Fig. 1), accessible to the encoder module (**accessible to root cache**, element 111, Fig. 1), for storing the contents of one or more data blocks previously transmitted by the encoder module (**a leaf cache for caching for previously transmitted web objects**, page 16, lines 16-20) wherein the encoder module (**root cache**, element 111, Fig. 1) determines whether there is a match between the contents of each of the extracted data blocks and the contents of at least one previously transmitted data block (**root cache determines whether the extracted web object is present in the leaf cache**, page 16, lines 22-27),

wherein responsive to said match (**if web object is present in the leaf cache**, page 16, lines 22-30), the encoder module encodes the respective extracted data block (**the web object is encoded as object signature, page 16**, lines 22-30) and transmits the respective extracted data block in encoded form (**root cache transmits the object signature**) to the at least one corresponding decoder module (**to the leaf cache**, page 16, lines 22-30);

responsive to no match (**if web object is not present at the leaf cache**, page 17, lines 1-5), the encoder module (**root cache**) transmits the respective extracted data block in intercepted form (**root cache transmits the actual web object**, page 17, lines 1-5) to the at least one corresponding decoder module (**to the leaf cache**, page 17, lines 9-11); and

each of the previously transmitted unique data blocks having a unique identifier (**each of the web objects having a URL as a unique identifier**, page 9, lines 20-26).

Malcolm does not explicitly disclose the memory comprises a least recently used data structure for storing one or more previously transmitted unique data blocks, said least recently used data structure having a maximum capacity and a position in an order of most recently used to least recently used of the one or more stored blocks, and

responsive to a match, associates the previously transmitted data block having the matching contents with the position in the least recently used data structure indicating the most recently used previously transmitted data block, and responsive to no match, the encoder module stores the extracted data block in the least recently used data structure, and associates the position of most recently used with the extracted data block.

However, Sarkissian discloses a least recently used data structure having a maximum capacity (LRU) (**CAM array/stack**, col. 31, lines 24-50) in a cache system of a packet network in which the addressable memory cells CAMs are ordered according to recentness of use, with the most recently used cache contents pointed to by the top CAM and the least recently used cache contents pointed to by the bottom CAM (col. 31, lines 41-50). Sarkissian also discloses when there is cache miss/no match, the contents of cache memory pointed to by the bottom CAM are replaced by the flow-entry from the flow-entry database 324. This now becomes the

most recently used entry, so the contents of the bottom CAM are moved to the top CAM and all CAM contents are shifted down (col. 19, lines 63-67, col. 20 , lines 1-2).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Sarkissian in employing a least recently used data structure/algorithm such that the leaf cache and root cache disclosed in Malcolm will implement the least recently used data structure of Sarkissian such that the leaf and root caches of Malcolm comprise a least recently used data structure for storing one or more previously transmitted unique data blocks, said least recently used data structure having a maximum capacity and a position in an order of most recently used to least recently used of the one or more stored blocks, and in response to no match, stores the extracted data block in the least recently used data structure, and associates the position of most recently used with the extracted data block.

The motivation to do so is to employ the least recently used replacement policy in cache system it replaces the least recently used flow entry when a cache replacement is needed because it is likely that a packet following a recent packet will belong to the same flow and thus the signature of a new packet will likely match a recently used flow record, and it is not highly likely that a packet associated with the least recently used flow-entry will arrive soon.

Regarding claim 24, Malcolm discloses the system of claim 2 wherein the memory stores a data structure for associating a signature with one or more of the previously transmitted data blocks (**root cache associates the object signature to the web object transmitted from the server device**, page 9, lines 3-8), and wherein the encoder module computes a signature for the

respective extracted data block (**root cache computes object signature for the web object**, page 9, lines 3-8), compares the computed signature with at least one signature associated with the one or more previously transmitted data blocks (**compares the object signature**), and responsive to a match in signature (**if there is a match**), selecting the one or more previously transmitted data blocks having the match in signature for content comparison with the respective extracted data block (**selecting the web object to compare it with the bitmap 114 stored for the web object**, page 16, lines 22-28).

Regarding claim 25, Malcolm discloses all the aspects of the claimed invention set forth in the rejection of claim 24 above, except fails to disclose the system of claim 24 wherein the data structure for associating a signature with one or more of the previously transmitted data blocks is a hash table having one or more bins.

However, Sarkissian discloses a hash structure is generated to associate a flow signature with a data flow and the signature being a bin number of the hash structure (col. 9, lines 22-28, col. 15, lines 65-67, col. 16, lines 1-7, col. 22, lines 25-26 and Fig. 8).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Sarkissian in employing a hash data structure to associate a flow signature with a data flow such that the data structure for associating a signature with one or more of the previously transmitted data blocks is a hash table having one or more bins.

The motivation to do so is to allow rapidly identifying and table look-up for a flow that may have a particular signature from a database of known flows.

Regarding claim 26, Malcolm discloses a system for encoding one or more repetitive data blocks in data communicated over a network comprising:

the encoder module intercepting the data (**root cache receives web object from server device**, page 16, lines 10-15 and Fig. 1) and extracting data blocks (**extracting object signatures**, page 16, lines 13-14) from different communication sessions (**from communication requests using different protocols such as FTP and HTTP**, page 7, lines 28-32) from the intercepted data (**from the web object**), each extracted block (**web object**, page 10-15) having a destination address supported for decoding by at least one corresponding decoder module (**each web object having a URL address**, page 9, lines 20-26), and the encoder module passing through data not having a supported destination address (**router switch 113 of root cache 111 passes through other types of requests other than the supported web object's URL identifier**, page 7, lines 28-32).

a memory (**a leaf cache**, element 111, Fig. 1), accessible to the encoder module (**accessible to root cache**, element 111, Fig. 1), for storing the contents of one or more data blocks previously transmitted by the encoder module (**a leaf cache for caching for previously transmitted web objects**, page 16, lines 16-20),

wherein the encoder module (**root cache**, element 111, Fig. 1) determines whether there is a match between the contents of each of the extracted data blocks and the contents of at least one previously transmitted data block (**root cache determines whether the extracted web object is present in the leaf cache**, page 16, lines 22-27) by computing a signature for the respective extracted data block and comparing the computed signature with at least one signature

associated with the one or more previously transmitted data blocks (**the root cache determines an object signature for the web object**, page 16, lines 13-14),

wherein, responsive to a match in signature (**if the object signature matches**, page 16, lines 22-30), the encoder module selects the one or more previously transmitted data blocks having the match in signature for content comparison with the respective data blocks (**root cache compares the bit map of the web object**, page 16, lines 22-27) and transmits the respective extracted data block in encoded form (**root cache transmits the object signature**) to the at least one corresponding decoder module (**to the leaf cache**, page 16, lines 22-30);

wherein, responsive to no match (**if web object is not present at the leaf cache**, page 17, lines 1-5), the encoder module (**root cache**) transmits the respective extracted data block in intercepted form (**root cache transmits the actual web object**, page 17, lines 1-5) to the at least one corresponding decoder module (**to the leaf cache**, page 17, lines 9-11).

Malcolm does not explicitly disclose the data structure for associating a signature with one or more of the previously transmitted data blocks is a hash table having one or more bins and the computed signature value being less than the number of hash table bins.

However, Sarkissian discloses a hash structure is generated to associate a flow signature with a data flow and the signature being a bin number of the hash structure (col. 9, lines 22-28, col. 15, lines 65-67, col. 16, lines 1-7, col. 22, lines 25-26 and Fig. 8; hash structure has a certain number of bins and each signature corresponds to one bin).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Sarkissian in employing a hash data structure to associate a flow signature with a data flow such that the

data structure for associating a signature with one or more of the previously transmitted data blocks is a hash table having one or more bins and the signature value is less than the number of hash table bins.

The motivation to do so is to allow rapidly identifying and table look-up for a flow that may have a particular signature from a database of known flows.

Regarding claim 27, Malcolm discloses a system for encoding one or more repetitive data blocks in data communicated over a network comprising:

the encoder module intercepting the data (**root cache receives web object from server device**, page 16, lines 10-15 and Fig. 1) and extracting data blocks (**extracting object signatures**, page 16, lines 13-14) from different communication sessions (**from communication requests using different protocols such as FTP and HTTP**, page 7, lines 28-32) from the intercepted data (**from the web object**), each extracted block (**web object**, page 10-15) having a destination address supported for decoding by at least one corresponding decoder module (**each web object having a URL address**, page 9, lines 20-26), and the encoder module passing through data not having a supported destination address (**router switch 113 of root cache 111 passes through other types of requests other than the supported web object's URL identifier**, page 7, lines 28-32).

a memory (**a leaf cache**, element 111, Fig. 1), accessible to the encoder module (**accessible to root cache**, element 111, Fig. 1), for storing the contents of one or more data blocks previously transmitted by the encoder module (**a leaf cache for caching for previously transmitted web objects**, page 16, lines 16-20),

wherein the encoder module (**root cache**, element 111, Fig. 1) determines whether there is a match between the contents of each of the extracted data blocks and the contents of at least one previously transmitted data block (**root cache determines whether the extracted web object is present in the leaf cache**, page 16, lines 22-27) by computing a signature for the respective extracted data block and comparing the computed signature with at least one signature associated with the one or more previously transmitted data blocks (**the root cache determines an object signature for the web object**, page 16, lines 13-14),

wherein, responsive to a match in signature (**if the object signature matches**, page 16, lines 22-30), the encoder module selects the one or more previously transmitted data blocks having the match in signature for content comparison with the respective data blocks (**root cache compares the bit map of the web object**, page 16, lines 22-27) and transmits the respective extracted data block in encoded form (**root cache transmits the object signature**) to the at least one corresponding decoder module (**to the leaf cache**, page 16, lines 22-30);

wherein, responsive to no match (**if web object is not present at the leaf cache**, page 17, lines 1-5), the encoder module (**root cache**) transmits the respective extracted data block in intercepted form (**root cache transmits the actual web object**, page 17, lines 1-5) to the at least one corresponding decoder module (**to the leaf cache**, page 17, lines 9-11).

Malcolm does not explicitly disclose the computed signature value being a modulo of the number of bins.

However, Sarkissian discloses a hash structure is generated to associate a flow signature with a data flow and the signature being a bin number of the hash structure (col. 9, lines 22-28,

col. 15, lines 65-67, col. 16, lines 1-7, col. 22, lines 25-26 and Fig. 8; hash structure has a certain number of bins and each signature corresponds to one bin).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Sarkissian in employing a hash data structure to associate a flow signature with a data flow such that the data structure for associating a signature with one or more of the previously transmitted data blocks is a hash table having one or more bins and the computed signature value being a modulo of the number of bins.

The motivation to do so is to allow rapidly identifying and table look-up for a flow that may have a particular signature from a database of known flows.

Regarding claim 28, Malcolm discloses the system of claim 2 further comprising an encapsulation module for encapsulating the extracted data blocks for transport over the network (**the server device encapsulates web object with object signature for transport over the network**, page 9, lines 3-8).

Regarding claim 29, Malcolm discloses the system of claim 28 wherein at least one of the extracted data blocks is included in a packet, and the packet is encapsulated as one packet (**web object is included in a MPEG packet**, page 6, lines 4-21).

Regarding claim 30, Malcolm discloses the system of claim 28 wherein at least one of the extracted data blocks is included in a packet (**web object is included in a MPEG packet**, page

6, lines 4-21) and the packet is encapsulated with at least one other packet in an outgoing packet for transmission (**web object is encapsulated with object signature for transport over the network**, page 9, lines 3-8).

Regarding claim 31, Malcolm discloses the system of claim 28 wherein the encapsulation of at least one of the extracted data blocks is transparent to one or more nodes in the network (**a cache may discard a web object that the object signature it receives can no longer be used to determine the web object, meaning the object signature is transparent to that cache**, page 12, lines 23-27).

Regarding claims 42, Malcolm discloses a system for decoding one or more repetitive data blocks in data communicated over a network comprising:

an decoder module (**a leaf cache**, element 111, Fig. 1), coupled in the network (**coupled in a communication network**, element 100, Fig. 1), the decoder module intercepting the data (**leaf cache receives object signature from root cache**, page 16, lines 29-30 and Fig. 1), the decoder module receiving data blocks (**object signatures**, page 16, lines 13-14) for different communication sessions (**for communication requests using different protocols such as FTP and HTTP**, page 7, lines 28-32) from a corresponding encoder module (**from the root cache**); and

a memory (**a memory storage**, element 112, Fig. 1), accessible to the decoder module (**accessible to leaf cache**, element 111, Fig. 1), for storing the contents of one or more data blocks previously transmitted by the encoder module (**for caching for previously**

**transmitted object signatures**, page 17, lines 1-4) wherein the decoder module (**leaf cache,** element 111, Fig. 1) determines whether the contents of each of the received data blocks is in encoded form (**leaf cache determines whether the received data is in object signature form**, page 17, lines 1-4),

wherein responsive to the respective data block being in encoded form (**if web object is in object signature form is received in the leaf cache**, page 16, lines 22-30), the decoder module selects the contents of a matching previously received block as the contents of the respective encoded block and the respective extracted data block (**leaf cache selects the web object of the matching object signature,** page 17, lines 1-4);

each of the previously transmitted unique data blocks having a unique identifier (**each of the web objects having a URL as a unique identifier**, page 9, lines 20-26).

wherein responsive to the data block being unencoded, the decoder module stores the contents of the respective received data block as a previously received data block (**when the actual web object is encountered at the leaf cache, the leaf cache stores it to be served to requesting client device**, page17, lines 9-14).

Malcolm does not explicitly disclose the memory comprises a least recently used data structure for storing one or more previously transmitted unique data blocks, said least recently used data structure having a maximum capacity and each of the previously transmitted unique data blocks having a unique identifier and a position in an order of most recently used to least recently used of the one or more stored blocks, and

wherein responsive to the respective data block being in encoded form, associates the previously transmitted data block having the matching contents with the position in the least recently used data structure indicating the most recently used previously transmitted data block.

However, Sarkissian discloses a least recently used data structure having a maximum capacity (LRU) (**CAM array/stack**, col. 31, lines 24-50) in a cache system of a packet network in which the addressable memory cells CAMs are ordered according to recentness of use, with the most recently used cache contents pointed to by the top CAM and the least recently used cache contents pointed to by the bottom CAM (col. 31, lines 41-50). Sarkissian further discloses when there is a cache hit/match, the contents of the CAM that produced the hit are put in the top CAM of the stack while the CAM contents of the CAM above the CAM that produced the hit are shifted down to fill the gap (col. 31, lines 32-40).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the decoding method of Malcolm with the teaching of Sarkissian in employing a least recently used data structure/algorithm such that the leaf cache and root cache disclosed in Malcolm will implement the least recently used data structure of Sarkissian such that the leaf and root caches of Malcolm comprise a least recently used data structure for storing one or more previously transmitted unique data blocks, said least recently used data structure having a maximum capacity and a position in an order of most recently used to least recently used of the one or more stored blocks, and in response to the respective data block being in encoded form, associates the previously transmitted data block having the matching contents with the position in the least recently used data structure indicating the most recently used previously transmitted data block.

The motivation to do so is to employ the least recently used replacement policy in cache system it replaces the least recently used flow entry when a cache replacement is needed because it is likely that a packet following a recent packet will belong to the same flow and thus the signature of a new packet will likely match a recently used flow record, and it is not highly likely that a packet associated with the least recently used flow-entry will arrive soon.

Regarding claim 43, Malcolm discloses the system of claim 42 wherein the decoder module (**a leaf cache**, element 111, Fig. 1) is coupled via a switch (**a router-switch**, element 113, Fig. 1) in a physical connection between two nodes of the network (**in a physical connection between a client and a server in a communication network**, elements 120. 130, Fig. 1), responsive to a first configuration of the switch (**responsive to the configuration of processing FTP or HTTP requests**), the encoder module processing data that traverse the physical connection between these two nodes (**the leaf cache processes FTP or HTTP requests**, page 7, lines 28-32), and responsive to a second configuration of the switch (**responsive to the configuration of processing all other types of requests**), the data bypassing the encoder module (**the cache passes through these data unchanged**, page 7, lines 28-32).

Regarding claims 45, Malcolm discloses the system of claim 42 wherein the decoder module, receives an indicator for identifying that the contents of a respective data block have been previously transmitted (**receives a object signature from the root cache for identifying the web object**, page 16, lines 29-30).

Regarding claim 46, Malcolm discloses the system of claim 45 wherein the indicator is a special symbol (**object signature is a special symbol, which is a form of dictionary compression**, page 12, lines 29-32).

Regarding claim 47, Malcolm discloses the system of claim 45 wherein the indicator is an extra header (**object signature is a form of dictionary compression of a web object**, page 12, lines 29-32).

Regarding claim 48, Malcolm discloses the system of claim 42 wherein at least one respective data block is a packet payload (**web object includes text or multimedia data**, page 6, lines 4-21).

Regarding claim 49, Malcolm discloses the system of claim 42 wherein at least one respective data block is a portion of a packet payload (**web object is a portion of the packet payload**, page 6, lines 4-21).

Regarding claim 50, Malcolm discloses the system of claim 42 wherein the decoder module decodes at least one extracted data block (**the leaf cache decodes the object signature**) using a synchronization mechanism for verifying the identification of the one or more previously received data blocks from a corresponding encoder module (**using this object signature as a synchronization mechanism to verify the identification of a previously received web object from a root cache**, page 16, lines 29-30, page 17, lines 1-4).

Regarding claim 51, Malcolm discloses the system of claim 50 wherein the synchronization mechanism is an explicit synchronization mechanism (**root cache receives an object signature from server device**, page 9, lines 3-8).

Regarding claim 58, Malcolm does not explicitly disclose the decoder module, responsive to no match in contents between the respective received data block and any one of the previously received data blocks, stores the respective received data block in the least recently used data structure and associates the position of most recently used with the respective received data block.

However, Sarkissian discloses when there is cache miss/no match, the contents of cache memory pointed to by the bottom CAM are replaced by the flow-entry from the flow-entry database 324. This now becomes the most recently used entry, so the contents of the bottom CAM are moved to the top CAM and all CAM contents are shifted down (col. 19, lines 63-67, col. 20 , lines 1-2).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Sarkissian in employing a least recently used data structure/algorithm such that the leaf cache and root cache disclosed in Malcolm will implement the least recently used data structure of Sarkissian such that the leaf and root caches of Malcolm comprise a least recently used data structure for storing one or more previously transmitted unique data blocks, said least recently used data structure having a maximum capacity and a position in an order of most recently used to least

recently used of the one or more stored blocks, and in response to no match, stores the extracted data block in the least recently used data structure, and associates the position of most recently used with the extracted data block.

The motivation to do so is to employ the least recently used replacement policy in cache system it replaces the least recently used flow entry when a cache replacement is needed because it is likely that a packet following a recent packet will belong to the same flow and thus the signature of a new packet will likely match a recently used flow record, and it is not highly likely that a packet associated with the least recently used flow-entry will arrive soon.

Regarding claims 89-91, Malcolm does not explicitly disclose the memory comprises a least recently used data structure for storing one or more previously received unique data blocks, said least recently used data structure having a maximum capacity and each of the previously received unique data blocks having a unique identifier and a position in an order of most recently used to least recently used of the one or more stored blocks, and

wherein responsive to the respective data block being in encoded form, associates the previously transmitted data block having the matching contents with the position in the least recently used data structure indicating the most recently used previously transmitted data block.

However, Sarkissian discloses a least recently used data structure having a maximum capacity (LRU) (**CAM array/stack**, col. 31, lines 24-50) in a cache system of a packet network in which the addressable memory cells CAMs are ordered according to recentness of use, with the most recently used cache contents pointed to by the top CAM and the least recently used cache contents pointed to by the bottom CAM (col. 31, lines 41-50). Sarkissian further discloses

when there is a cache hit/match, the contents of the CAM that produced the hit are put in the top CAM of the stack while the CAM contents of the CAM above the CAM that produced the hit are shifted down to fill the gap (col. 31, lines 32-40).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the decoding method of Malcolm with the teaching of Sarkissian in employing a least recently used data structure/algorithm such that the leaf cache and root cache disclosed in Malcolm will implement the least recently used data structure of Sarkissian such that the leaf and root caches of Malcolm comprise a least recently used data structure for storing one or more previously transmitted unique data blocks, said least recently used data structure having a maximum capacity and a position in an order of most recently used to least recently used of the one or more stored blocks, and in response to the respective data block being in encoded form, associates the previously transmitted data block having the matching contents with the position in the least recently used data structure indicating the most recently used previously transmitted data block.

The motivation to do so is to employ the least recently used replacement policy in cache system it replaces the least recently used flow entry when a cache replacement is needed because it is likely that a packet following a recent packet will belong to the same flow and thus the signature of a new packet will likely match a recently used flow record, and it is not highly likely that a packet associated with the least recently used flow-entry will arrive soon.

6. Claims 5, 13-14, 44, 52-54, 83-84 are rejected under 35 U.S.C. 103(a) as being unpatentable over Malcolm in view of Sarkissian, and in further view of Garcia-Luna-Aceves (US Publication 2002/0056416).

Regarding claims 5, 44, the combined system of Malcolm and Sarkissian discloses all the aspects of the claimed invention set forth in the rejection of claim 2 above, except fails to explicitly disclose the encoder deciding a route for the respective extracted data block to the at least one corresponding decoder module supporting its destination address.

Malcolm does not explicitly show the encoder module receives routing information and routing criteria over the network from each of one or more corresponding decoder modules with which it communicates and determines the one or more addresses supported by each respective decoder module from the routing information.

However, Garcia-Luna-Aceves discloses web client receives routing information/criteria such as routing delay and routing distance between a web router and the client, and that the web router determines update network topology information and which of the other web routers support routing for the web client request based on the routing criteria (paragraphs 0043, 0127, 0137).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Garcia-Luna-Aceves in determining and forwarding routing information from a web router to a web client such that the root cache of Malcolm will decide a route and forward routing information and routing criteria over the network to the leaf cache.

The motivation to do so is to enable the selection of the optimal route of web routers to service client information object request.

Regarding claims 13-14, the combined system of Malcolm and Sarkissian does not explicitly disclose the system of claim 11 wherein the synchronization mechanism is an implicit synchronization and the implicit synchronization mechanism is a reliable network transport protocol.

However, Garcia-Luna-Aceves discloses the topology information passed between web routers are transported using reliable transmission protocol to provide implicit synchronization (paragraph 0043).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the system and method of Malcolm and Sarkissian with the teaching of Garcia-Luna-Aceves in using a reliable transport communication protocol to provide implicit synchronization when updating network topology information. The motivation to do so is to provision reliable transmissions between web routers.

Regarding claims 52-54, the combined system of Malcolm and Sarkissian does not explicitly disclose the system wherein the synchronization mechanism is an implicit synchronization and the implicit synchronization mechanism is a reliable network transport protocol.

However, Garcia-Luna-Aceves discloses the topology information passed between web routers are transported using reliable transmission protocol to provide implicit synchronization

(paragraph 0043). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the system and method of Malcolm and Sarkissian with the teaching of Garcia-Luna-Aceves in using a reliable transport communication protocol to provide implicit synchronization when updating network topology information. The motivation to do so is to provision reliable transmissions between web routers.

Regarding claims 83-84, the combined system of Malcolm and Sarkissian does not explicitly show the encoder module receives routing information and routing criteria over the network from each of one or more corresponding decoder modules with which it communicates and determines the one or more addresses supported by each respective decoder module from the routing information.

However, Garcia-Luna-Aceves discloses web client receives routing information/criteria such as routing delay and routing distance between a web router and the client, and that the web router determines update network topology information and which of the other web routers support routing for the web client request based on the routing criteria (paragraphs 0043, 0127, 0137).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm and Sarkissian with the teaching of Garcia-Luna-Aceves in determining and forwarding routing information from a web router to a web client such that the leaf cache of Malcolm will receive routing information and routing criteria over the network from the root cache with which it communicates and determines the one or more addresses supported by the root cache based on the routing information.

The motivation to do so is to enable the selection of the optimal route of web routers to service client information object request.

7. Claims 19, 55, 59, 92 are rejected under 35 U.S.C. 103(a) as being unpatentable over the admitted prior art, Malcolm et al. (WO 00/07124), in view of Sarkissian et al. (USP 6,771,646), and in further view of Storer (USP 4,876,541).

Regarding claim 19, the combined system of Malcolm and Sarkissian does not explicitly show that in response to the least recently used data structure being at the maximum capacity, the encoder module deletes the previously transmitted data block having the order position of the least recently used data block.

However, Storer discloses when the least recently used data structure is full, the data block being at the position of the least recently used data block is deleted (col. 16, lines 66-68, col. 17, lines 1-6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combined system and method of Malcolm and Sarkissian with the teaching of Storer in employing a least recently used data structure/algorithm such that the least recently used data structure of the combined system Malcolm and Sarkissian will delete the previously transmitted data block having the order position of the least recently used data block in response to the least recently used data structure being at the maximum capacity.

The motivation to do so is to create new space for the new flow entry by removing a certain number of the least recently used data blocks at the end of the queue.

Regarding claim 55, the combined system of Malcolm and Sarkissian does not explicitly disclose the system of claim 42 wherein responsive to the data block being unencoded, the decoder module stores the contents of the respective received data block as a previously received data block further comprises determining whether to delete at least one of the previously received data blocks.

However, Storer discloses when the least recently used data structure is full, the data block being at the position of the least recently used data block is deleted (col. 16, lines 66-68, col. 17, lines 1-6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combined system and method of Malcolm and Sarkissian with the teaching of Storer in employing a least recently used data structure/algorithm such that the least recently used data structure of the combined system Malcolm and Sarkissian will delete the previously transmitted data block having the order position of the least recently used data block in response to the least recently used data structure being at the maximum capacity.

The motivation to do so is to create new space for the new flow entry by removing a certain number of the least recently used data blocks at the end of the queue.

Regarding claim 59, the combined system of Malcolm and Sarkissian does not explicitly show that in response to the least recently used data structure being at the maximum capacity, the encoder module deletes the previously transmitted data block having the order position of the least recently used data block.

However, Storer discloses when the least recently used data structure is full, the data block being at the position of the least recently used data block is deleted (col. 16, lines 66-68, col. 17, lines 1-6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combined system and method of Malcolm and Garcia-Luna-Aceves with the teaching of Storer in employing a least recently used data structure/algorithm such that the least recently used data structure of the combined system Malcolm and Garcia-Luna-Aceves delete the previously transmitted data block having the order position of the least recently used data block in response to the least recently used data structure being at the maximum capacity.

The motivation to do so is to create new space for the new flow entry by removing a certain number of the least recently used data blocks at the end of the queue.

Regarding claim 92, the combined system and method of Malcolm and Sarkissian does not explicitly disclose in response to the least recently used data structure being at the maximum capacity, deleting the previously received data block having the order position of the least recently used data block,

However, Storer discloses when the least recently used data structure is full, the data block being at the position of the least recently used data block is deleted (col. 16, lines 66-68, col. 17, lines 1-6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combined system and method of Malcolm and Sarkissian with

the teaching of Storer in employing a least recently used data structure/algorithm such that the least recently used data structure of the combined system Malcolm and Sarkissian will delete the previously transmitted data block having the order position of the least recently used data block in response to the least recently used data structure being at the maximum capacity.

The motivation to do so is to create new space for the new flow entry by removing a certain number of the least recently used data blocks at the end of the queue.

8. Claims 32-33 are rejected under 35 U.S.C. 103(a) as being unpatentable over Malcolm in view of Sarkissian, and in further view of Gorman et al. (USP 5,394,879).

Regarding claims 32, 33, the combined system of Malcolm and Sarkissian discloses all the aspects of the claimed invention set forth in the rejection of claim 28 above, except fails to disclose the system of claim 28 wherein the encapsulation module comprises a timer mechanism for ensuring that the at least one extracted data block is held in a buffer coupled to the encapsulation module for no more than a pre-determined maximum time before being transmitted.

However, Gorman discloses using a timer to trigger the transmission of digital encoded signal (col. 14, lines 10-18, Fig. 6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combined system of Malcolm and Sarkissian with the teaching of Gorman such that a timer mechanism is used in the encoding method of Seitz for ensuring that the at least one extracted data block is held in a buffer coupled to the encapsulation module for no more than a pre-determined maximum time before being transmitted. The motivation to do so

is to enable error detection and correction of the encoded digital signal prior to its transmission to a receiver unit.

**9.** Claims 34-36, 63-68, 80-81 are rejected under 35 U.S.C. 103(a) as being unpatentable over Malcolm in view of Sarkissian, and in further view of Adriano et al. (USP 6,484,210).

Regarding claims 34-36, 80-81, the combined system of Malcolm and Sarkissian discloses all the aspects of the claimed invention set forth in the rejection of claim 28 above, except fails to disclose the system of claim 28 wherein the encoder module encodes at least one data block at a first layer of a model describing the flow of data across a network and the encapsulation module encapsulates the at least one extracted data block at a second layer of the model.

However, Adriano discloses encoding a data payload in a TCP packet (**first layer and connection-oriented layer**) and encapsulating the TCP packet in an UDP packet (**second layer and connectionless layer**, col. 10, lines 31-50 and Fig. 5; TCP and UDP are the same layer).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combined system and method of Malcolm and Sarkissian with the teaching of Adriano in using TCP and UDP to encode and encapsulate packets, respectively such that the root cache of Malcolm encodes at least one data block at a first layer of a model and the at least one extracted data block is encapsulated at a second layer of the model.

The motivation to do so is to route the entire TCP packet tunneled within the UDP packet to the destination IP address of the TCP packet.

Regarding claim 63, the combined system of Malcolm and Sarkissian does not explicitly show that a decapsulation module for decapsulating a block of data received (**decapsulator module for decapsulating UDP packet 500**, col. 10, lines 31-50, Fig. 5) over the network from a source address supported by a corresponding encoder module (**from a source address of an encoder module**, col. 10, lines 31-50 and Fig. 5).

However, Adriano discloses a system further comprises a decapsulation module for decapsulating a block of data received (**decapsulator module for decapsulating UDP packet 500**, col. 10, lines 31-50, Fig. 5) over the network from a source address supported by a corresponding encoder module (**from a source address of an encoder module**, col. 10, lines 31-50 and Fig. 5).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combined system and method of Malcolm and Sarkissian with the teaching of Adriano such that the decapsulation module of Malcolm decapsulates UDP packet using the IP source address of the encoder module.

The motivation to do so is to identify the source from which the packet is originated.

Regarding claims 64-68, Malcolm discloses the system of claim 63 wherein the received data block is included in a packet (**web object is included in a MPEG packet**, page 6, lines 4-21). Malcolm does not explicitly disclose the packet has been decapsulated as one packet from inside another packet and decapsulating the received block of data of a model describing the flow of data across a network and decoding the received block of data at a second layer of the model, first layer and second layer being the same layer.

However, Adriano discloses a system that comprises a decapsulation module for decapsulating a block of data received (**decapsulator module for decapsulating UDP packet 500**, col. 10, lines 31-50, Fig. 5) over the network from a source address supported by a corresponding encoder module (**from a source address of an encoder module**, col. 10, lines 31-50 and Fig. 5).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combined system and method of Malcolm and Sarkissian with the teaching of Adriano such as decapsulating TCP packet 555 from within a UDP packet 500 (TCP and UDP are the same layer and TCP is a connection-oriented layer and UDP is a connectionless layer) such that the packet disclosed in Malcolm will be decapsulated as one TCP packet from inside a UDP packet.

The motivation to do so is to route the entire TCP packet tunneled within the UDP packet to the destination IP address of the TCP packet.

10. Claims 37-41, 69-71 are rejected under 35 U.S.C. 103(a) as being unpatentable over the admitted prior art, Malcolm et al. (WO 00/07124), in view of Garcia-Luna-Aceves (US Publication 2001/0056416)

Regarding claims 37-41, Malcolm discloses a system and method for encoding one or more repetitive data blocks in data communicated over a network comprising:

an encoder module (**a root cache**, element 111, Fig. 1), coupled in the network (**coupled in a communication network**, element 100, Fig. 1), the encoder module intercepting the data (**root cache receives web object from server device**, page 16, lines 10-15 and Fig. 1),

the encoder module extracting data blocks (**object signatures**, page 16, lines 13-14) from different communication sessions (**from communication requests using different protocols such as FTP and HTTP**, page 7, lines 28-32) from the intercepted data (**from the web object**), each extracted block (**web object**, page 10-15) having a destination address supported for decoding by at least one corresponding decoder module (**each web object having a URL address**, page 9, lines 20-26), and the encoder module passing through data not having a supported destination address (**router switch 113 of root cache 111 passes through other types of requests other than the supported web object's URL identifier**, page 7, lines 28-32).

a memory (**a leaf cache**, element 111, Fig. 1), accessible to the encoder module (**accessible to root cache**, element 111, Fig. 1), for storing the contents of one or more data blocks previously transmitted by the encoder module (**a leaf cache for caching for previously transmitted web objects**, page 16, lines 16-20),

wherein the encoder module (**root cache**, element 111, Fig. 1) determines whether there is a match between the contents of each of the extracted data blocks and the contents of at least one previously transmitted data block (**root cache determines whether the extracted web object is present in the leaf cache**, page 16, lines 22-27),

wherein responsive to said match (**if web object is present in the leaf cache**, page 16, lines 22-30), the encoder module encodes the respective extracted data block (**the web object is encoded as object signature, page 16**, lines 22-30) and transmits the respective extracted data block in encoded form (**root cache transmits the object signature**) to the at least one corresponding decoder module (**to the leaf cache**, page 16, lines 22-30);

responsive to no match (**if web object is not present at the leaf cache**, page 17, lines 1-5), the encoder module (**root cache**) transmits the respective extracted data block in intercepted form (**root cache transmits the actual web object**, page 17, lines 1-5) to the at least one corresponding decoder module (**to the leaf cache**, page 17, lines 9-11).

Malcolm does not explicitly show the encoder module receives routing information and routing criteria over the network from each of one or more corresponding decoder modules with which it communicates and determines the one or more addresses supported by each respective decoder module from the routing information.

However, Garcia-Luna-Aceves discloses web client receives routing information/criteria such as routing delay and routing distance between a web router and the client, and that the web router determines update network topology information and which of the other web routers support routing for the web client request based on the routing criteria (paragraphs 0043, 0127, 0137).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Garcia-Luna-Aceves in determining and forwarding routing information from a web router to a web client such that the leaf cache of Malcolm will receive routing information and routing criteria over the network from the root cache with which it communicates and determines the one or more addresses supported by the root cache based on the routing information.

The motivation to do so is to enable the selection of the optimal route of web routers to service client information object request.

Regarding claim 69, Malcolm discloses a system for decoding one or more repetitive data blocks in data communicated over a network comprising:

an decoder module (**a leaf cache**, element 111, Fig. 1), coupled in the network (**coupled in a communication network**, element 100, Fig. 1); and

a memory (**a memory storage**, element 112, Fig. 1), accessible to the decoder module (**accessible to leaf cache**, element 111, Fig. 1), for storing the contents of one or more data blocks previously received from the encoder module (**for caching for previously transmitted object signatures**, page 17, lines 1-4),

wherein the decoder module (**leaf cache**, element 111, Fig. 1) determines whether the contents of each of the received data blocks is in encoded form (**leaf cache determines whether the received data is in object signature form**, page 17, lines 1-4),

wherein responsive to the respective data block being in encoded form (**if web object is in object signature form is received in the leaf cache**, page 16, lines 22-30), the decoder module selects the contents of a matching previously received block as the contents of the respective encoded block and the respective extracted data block (**leaf cache selects the web object of the matching object signature,** page 17, lines 1-4);

wherein responsive to the data block being unencoded, the decoder module stores the contents of the respective received data block as a previously received data block (**when the actual web object is encountered at the leaf cache, the leaf cache stores it to be served to requesting client device**, page 17, lines 9-14).

Malcolm does not explicitly disclose the decoder module transmits routing information for identifying one or more addresses it supports over the network to the encoder module.

However, Garcia-Luna-Aceves discloses web client receives from web router the routing information/criteria such as routing delay and routing distance between a web router and the client, and that the web router determines update network topology information and which of the other web routers support routing for the web client request based on the routing criteria (paragraphs 0043, 0127, 0137).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Garcia-Luna-Aceves in determining and forwarding routing information from a web router to a web client such that the root cache of Malcolm will transmit routing information and routing criteria over the network to the leaf cache.

The motivation to do so is to enable the selection of the optimal route of web routers to service client information object request.

Regarding claim 70, Malcolm does not disclose the decoder module participates in one or more routing protocols for obtaining routing information.

However, Garcia-Luna-Aceves discloses web routers use routing information provided by routing protocols such as OSPF and BGP (paragraph 0083).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Garcia-Luna-Aceves in determining and forwarding routing information from a web router to a web client using different routing protocols such as OSPF and BGP.

The motivation to do so is to web routers to derive distances to client address ranges.

Regarding claim 71, Malcolm discloses a system for encoding one or more repetitive data blocks in data communicated over a network comprising:

an encoder module (**a root cache**, element 111, Fig. 1), coupled in the network (**coupled in a communication network**, element 100, Fig. 1), the encoder module intercepting the data (**root cache receives web object from server device**, page 16, lines 10-15 and Fig. 1), the encoder module extracting data blocks (**object signatures**, page 16, lines 13-14) from different communication sessions (**from communication requests using different protocols such as FTP and HTTP**, page 7, lines 28-32) from the intercepted data (**from the web object**), each extracted block (**web object**, page 10-15) having a destination address supported for decoding by at least one corresponding decoder module (**each web object having a URL address**, page 9, lines 20-26), and the encoder module passing through data not having a supported destination address (**router switch 113 of root cache 111 passes through other types of requests other than the supported web object's URL identifier**, page 7, lines 28-32).

a first memory (**a leaf cache**, element 111, Fig. 1), accessible to the encoder module (**accessible to root cache**, element 111, Fig. 1), for storing the contents of one or more data blocks previously transmitted by the encoder module (**a leaf cache for caching for previously transmitted web objects**, page 16, lines 16-20) wherein the encoder module (**root cache**, element 111, Fig. 1) determines whether there is a match between the contents of each of the extracted data blocks and the contents of at least one previously transmitted data block (**root cache determines whether the extracted web object is present in the leaf cache**, page 16, lines 22-27),

wherein responsive to said match (**if web object is present in the leaf cache**, page 16, lines 22-30), the encoder module encodes the respective extracted data block (**the web object is encoded as object signature, page 16**, lines 22-30) and transmits the respective extracted data block in encoded form (**root cache transmits the object signature**) to the at least one corresponding decoder module (**to the leaf cache**, page 16, lines 22-30), and responsive to no match (**if web object is not present at the leaf cache**, page 17, lines 1-5), the encoder module (**root cache**) transmits the respective extracted data block in intercepted form (**root cache transmits the actual web object**, page 17, lines 1-5) to the at least one corresponding decoder module (**to the leaf cache**, page 17, lines 9-11); and

the at least one corresponding decoder module (**the leaf cache**, element 111, Fig. 1) being coupled in the network (**coupled in a communication network**, element 100, Fig. 1), the decoder module receiving data blocks for different communication sessions from the encoder module (**leaf cache receiving object signature from root cache for communication requests using different protocols such as FTP and HTTP**, page 7, lines 28-32); and

a memory (**a memory storage**, element 112, Fig. 1), accessible to the decoder module (**accessible to leaf cache**, element 111, Fig. 1), for storing the contents of one or more data blocks previously received from the encoder module (**for caching for previously transmitted object signatures**, page 17, lines 1-4),

wherein the decoder module (**leaf cache**, element 111, Fig. 1) determines whether the contents of each of the received data blocks is in encoded form (**leaf cache determines whether the received data is in object signature form**, page 17, lines 1-4),

wherein responsive to the respective data block being in encoded form (**if web object is in object signature form is received in the leaf cache**, page 16, lines 22-30), the decoder module selects the contents of a matching previously received block as the contents of the respective encoded block and the respective extracted data block (**leaf cache selects the web object of the matching object signature,** page 17, lines 1-4);

wherein responsive to the data block being unencoded, the decoder module stores the contents of the respective received data block as a previously received data block (**when the actual web object is encountered at the leaf cache, the leaf cache stores it to be served to requesting client device**, page 17, lines 9-14).

Malcolm does not explicitly disclose the decoder module transmits routing information for identifying one or more addresses it supports over the network to the encoder module.

However, Garcia-Luna-Aceves discloses web client receives from web router the routing information/criteria such as routing delay and routing distance between a web router and the client, and that the web router determines update network topology information and which of the other web routers support routing for the web client request based on the routing criteria (paragraphs 0043, 0127, 0137).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the encoding method of Malcolm with the teaching of Garcia-Luna-Aceves in determining and forwarding routing information from a web router to a web client such that the root cache of Malcolm will transmit routing information and routing criteria over the network to the leaf cache.

The motivation to do so is to enable the selection of the optimal route of web routers to service client information object request.

11. Claim 88 is rejected under 35 U.S.C. 103(a) as being unpatentable over Malcolm in view of Storer (4,876,541).

Regarding claim 88, Malcolm discloses the method of claim 85 further comprising responsive to the data block being unencoded, the decoder module stores the contents of the respective received data block as a previously received data block (**when the actual web object is encountered at the leaf cache, the leaf cache stores it to be served to requesting client device**, page17, lines 9-14).

Malcolm does not explicitly show determining whether to delete at least one of the previously received data blocks.

However, Storer discloses when the least recently used data structure is full, the data block being at the position of the least recently used data block is deleted (col. 16, lines 66-68, col. 17, lines 1-6).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combined system and method of Malcolm with the teaching of Storer in employing a least recently used data structure/algorithm such that the least recently used data structure of the combined system Malcolm will delete the previously transmitted data block having the order position of the least recently used data block in response to the least recently used data structure being at the maximum capacity.

The motivation to do so is to create new space for the new flow entry by removing a certain number of the least recently used data blocks at the end of the queue.

12. Claims 93-95 are rejected under 35 U.S.C. 103(a) as being unpatentable over Malcolm in view of Adriano et al. (6,484,210).

Regarding claims 93-95, the combined system of Malcolm does not explicitly disclose the packet has been decapsulated as one packet from inside another packet and decapsulating the received block of data of a model describing the flow of data across a network and decoding the received block of data at a second layer of the model, first layer and second layer being the same layer.

However, Adriano discloses a system further comprises a decapsulation module for decapsulating a block of data received (**decapsulator module for decapsulating UDP packet 500**, col. 10, lines 31-50, Fig. 5) over the network from a source address supported by a corresponding encoder module (**from a source address of an encoder module**, col. 10, lines 31-50 and Fig. 5).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the combined system and method of Malcolm with the teaching of Adriano such as decapsulating TCP packet 555 from within a UDP packet 500 (TCP and UDP are the same layer and TCP is a connection-oriented layer and UDP is a connectionless layer) such that the packet disclosed in Malcolm will be decapsulated as one TCP packet from inside a UDP packet.

The motivation to do so is to route the entire TCP packet tunneled within the UDP packet to the destination IP address of the TCP packet.

***Allowable Subject Matter***

13. Claims 20-23, 60-62 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

The following is a statement of reasons for the indication of allowable subject matter:

In claim 20, the system of claim 19 further comprising a synchronization mechanism including a same size for the least recently used data structure as the size of a second least recently used data structure accessible by the corresponding decoder module receiving the respective extracted data block, and a reliable network transport protocol being used for the transmission of the respective extracted data block.

In claim 21, the system of claim 19 further comprising a synchronization mechanism including an installed flag associated each of the previously transmitted data blocks in the least recently used data structure, the installed flag indicating whether the associated data block has been stored in a second least recently used data structure accessible by the corresponding decoder module that has received the respective extracted data block.

In claim 60, the system of claim 56 further comprising a synchronization mechanism including a same size for the least recently used data structure as the size of a second least recently used data structure accessible by the corresponding encoder module that transmitted the respective received data block, and a reliable network transport protocol being used for the transmission of the respective received data block.

In claim 61, the system of claim 56 further comprising a synchronization mechanism including a same size for the least recently used data structure as the size of a second least recently used data structure accessible by the corresponding encoder module, an indicator for each received data block indicating whether the received data block has been previously associated with the second least recently used data structure, and an installation acknowledgement transmitting by the decoder module to the corresponding encoder module responsive to an installation of each received block in the least recently used data structure.

In claim 62, the system of claim 56 further comprising a synchronization mechanism wherein the decoder module receives a version number of the previously received data block having the matching contents, the version number indicating how many times the identifier for this data block has been re-used.

***Response to Arguments***

14. Applicant's arguments with respect to claims 2-15, 18-55, 58-81, 83-95 have been considered but are moot in view of the new ground(s) of rejection.

*Conclusion*

15. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Kevin Mew whose telephone number is 571-272-3141. The examiner can normally be reached on 9:00 am - 5:30 pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Seema Rao can be reached on 571-272-3174. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Seema S. Rao
SEEMA S. RAO 5/15/06
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2600

Kevin Mew KM
Work Group 2616